This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims

1. (Currently Amended) A database engine comprising:

a transactional mechanism supporting heterogeneous distributed transactions, said transactional mechanism having:

means for recognizing data sources conforming to an industry standard interface for interaction between a transaction manager and resource manager, ~~the X/Open XA standards~~, said data sources including structured and non-structured external data sources; and

means for managing transactions in which said data sources participate by:

generating a global transaction identifier for each distributed transaction;

accessing support functions for data source types for data sources registered to participate in the transaction identified by the global transaction identifier; and

maintaining a list of all of the recognized heterogeneous data sources that participated in the global transaction identified by the global transaction ID.

2. (Original) The database engine according to claim 1, in which said transactional mechanism further comprises, for each of said data sources:

means for supporting transactional events conforming to the X/Open XA standards, said transactional events including prepare, commit, rollback, redo, and undo.

3. (Original) The database engine according to claim 1, further comprises:

support functions configured to support each recognized data source.

4. (Original) The database engine according to claim 3, further comprises:

means for invoking said support functions at appropriate transactional events, said transactional events including prepare, commit, rollback, redo, and undo.

5. (Original) The database engine according to claim 1, in which

said database engine supports at least one database application;

wherein each of said data sources has one or more instances; and

wherein said at least one database application interacts with said one or more instances via said database engine.

6. (Original) The database engine according to claim 1, wherein

each of said data sources is a resource manager assigned with a unique identifier.

7. (Currently Amended) The database engine according to claim 1, wherein said transactional mechanism further comprises:

~~means for generating and maintaining a global transaction ID for each of said heterogeneous distributed transactions; and~~

means for producing a 2-phase commit transaction model for said data sources.

8. (Canceled)

9. (Currently Amended) ~~A computer readable medium storing a computer program implementing a~~ The database engine of claim 1, ~~said computer program comprising computer-executable instructions for~~ wherein accessing the support functions comprises:

~~recognizing said data sources;~~

~~assigning each of said data sources with a unique identifier;~~

~~generating one or more instances for each of said data sources;~~

~~configuring support functions to support said data sources;~~

~~managing said transactions in which said data sources participate;~~

~~generating and maintaining a global transaction ID for each of said heterogeneous distributed transactions;~~

invoking said support functions at appropriate transactional events including begin, prepare, commit, rollback, redo, and undo; and

producing a 2-phase commit transaction model for said data sources.

10. (Currently Amended) A database server comprising:

a database engine comprising:

a transactional mechanism supporting heterogeneous distributed transactions, said transactional mechanism having:

means for recognizing data sources conforming to an industry standard interface for interaction between a transaction manager and resource manager ~~the X/Open XA standards~~, said data sources including structured and non-structured external data sources;

~~support functions configured to support each recognized data source;~~

means for managing transactions in which said data sources participate~~; and~~ by:

generating a global transaction identifier for each distributed transaction;

accessing support functions for data source types for data sources registered to participate in the transaction identified by the global transaction identifier; and

maintaining a list of all of the recognized heterogeneous data sources that participated in the global transaction identified by the global transaction ID

~~means for invoking said support functions at appropriate transaction events including prepare, commit, and rollback~~.

11. (Original)The database server according to claim 10, further comprising:

at least one database application;

wherein said database engine supports said at least one database application;

wherein each of said data sources has one or more instances; and

wherein said at least one database application interacts with said one or more instances via said database engine.

12. (Original) The database server according to claim 11, wherein each of said data sources is a resource manager assigned with a unique identifier.

13. (Currently Amended) The database server according to claim 10, further comprising:

~~means for generating and maintaining a global transaction ID for each of said heterogeneous distributed transactions; and~~

means for producing a 2-phase commit transaction model for said data sources.

14. (Canceled)

15. (Currently Amended) ~~A computer readable medium storing a computer program implementing the~~ The database server of claim 10, ~~said computer program comprising computer-executable instructions for~~ wherein accessing support functions comprises:

~~recognizing said data sources;~~

~~assigning each of said data sources with a unique identifier;~~

~~generating one or more instances for each of said data sources;~~

~~configuring support functions to support said data sources;~~

~~managing said transactions in which said data sources participate;~~

~~generating and maintaining a global transaction ID for each of said heterogeneous distributed transactions;~~

invoking said support functions at appropriate transactional events including begin, prepare, commit, rollback, redo, and undo; and

producing a 2-phase commit transaction model for said data sources.

16. (Currently Amended) A method of integrating a database system to support heterogeneous distributed transactions, comprising:

recognizing data sources conforming to an industry standard interface for interaction between a transaction manager and resource manager ~~the X/Open XA standards~~, said data sources including structured and non-structured data sources external to said database system; and

configuring a database engine with a transactional mechanism, said transactional mechanism managing said heterogeneous distributed transactions in which said data sources participate, wherein said transactional mechanism is capable of managing transactions in which said data sources participate by:

generating a global transaction identifier for each distributed transaction;

accessing support functions for data source types for data sources registered to participate in the transaction identified by the global transaction identifier; and

maintaining a list of all of the recognized heterogeneous data sources that participated in the global transaction identified by the global transaction ID

~~assigning each of said data sources with a unique identifier;~~

~~generating one or more instances for each of said data sources;~~

~~generating and maintaining a global transaction ID for each of said heterogeneous distributed transactions;~~

~~invoking support functions for said data sources at appropriate transactional events; and~~

~~producing a 2-phase commit transaction model supporting said heterogeneous distributed transactions with said data sources~~.

17. (Original) The method according to claim 16, further comprising:

constructing support functions for each of said data sources that participates in said heterogeneous distributed transactions.

18. (Original) The method according to claim 16,

wherein said transactional events conform to the X/Open XA standards; and

wherein said transactional events include begin, prepare, commit, rollback, redo, and undo.

19. (Canceled)

20. (Canceled)

21. (New) The database engine of claim 1, wherein the industry standard interface comprises the X/Open XA standard.

22. (New) The database server of claim 10, wherein the industry standard interface comprises the X/Open XA standard.

23. (New) The method of claim 16, wherein the industry standard interface comprises the X/Open XA standard.